Hans U. Simon
Malte Darnstädt

Bochum, den 9.06.2010
Abgabe bis zum 16.06. in der Übung

Übungen zur Vorlesung

Komplexitätstheorie

Sommer 2010

Übungsblatt 7

Aufgabe 7.1

Ein *Straight-Line-Programm* ist ein endliches Programm ohne Sprünge oder bedingte Anweisungen (wie "goto" oder "if"). Wir betrachten Straight-Line-Programme über einer einfachen Programmiersprache, die nur boolsche Operationen als Anweisungen kennt:

- Die Eingabe ist in Variablen gegeben: $x_1, \ldots, x_n \in \{0, 1\}$
- Die i-te Zeile besteht aus einer Anweisung der Form $y_i := a$ OP b oder $y_i := \neg a$ wobei $a, b \in \{0, 1, x_1, \ldots, x_n, y_1, \ldots, y_{i-1}\}$ und OP $\in \{\wedge, \vee\}$
- Die Ausgabe des Programms ist der Wert y_S aus der letzten Zeile

Sei $f : \{0,1\}^n \to \{0,1\}$ eine beliebige Abbildung und $S \in \mathbb{N}$. Zeige:
f kann von einem Schaltkreis der Größe S berechnet werden $\iff$ f kann von einem Straight-Line-Programm mit S Zeilen berechnet werden

Aufgabe 7.2

Zeige, dass entscheidbare Sprachen existieren, die in $\mathrm{P}_{/\mathrm{poly}}$ aber nicht in P liegen.

Aufgabe 7.3

Eine Sprache $L \subseteq \{0,1\}^*$ heißt *sparse* (dünn), falls es ein Polynom p gibt so dass für alle n gilt: $|L \cap \{0,1\}^n| \le p(n)$

Zeige: Jede dünne Sprache befindet sich in $\mathrm{P}_{/\mathrm{poly}}$.

Aufgabe 7.4

Eine Sprache $L \subseteq \{1\}^*$, also eine Sprache die nur aus Wörtern der Form 1^k besteht, heißt *unär.*

Zeige: Aus der Existenz einer unären, NP-vollständigen Sprache folgt P=NP.

Hinweis: Sei L die unäre, NP-vollständige Sprache und f die Reduktionsabbildung von SAT auf L. Betrachte einen Algorithmus, der SAT rekursiv entscheidet. Der Algorithmus benötigt im Allgemeinen exponentiell viele Verzweigungen. Wie kann man mit Hilfe von f den Rekursionsbaum verkleinern?